**Project: Online Food Delivery System**

|  |  |  |
| --- | --- | --- |
|  | **FUNCTIONAL SPECIFICATION** | |
|  |  |  |
| **Project Team:** |  | 4 |
| **Project Name:** |  | Online Food Delivery System |

**Table of Contents**

1. INTRODUCTION………………………………………………………………………………

2. SYSTEM OVERVIEW …………………….………………………………………………..…

3. SUB-SYSTEM DETAILS ……………...……………………………………………………...

4. TECHNOLOGY STACK…………………………………………………………………………….

5. ASSUMPTIONS…………………………………………………………………………………

6. DIAGRAMS……………………………………………………………………….....…………

7. ADVANTAGES…………………………………………………………………………………

8. OUTPUT SCREENSHOT FOR PROJECT……………………………………………………..

9. CONCLUSION………………………………………………………………………………….

**ABSTRACT**

An Online Food Ordering System is proposed here which simplifies the food ordering process. The proposed system shows a user interface and updates the menu with all available options so that it eases the customer’s work. Customers can choose more than one item to make an order and can view order details before logging off. The order confirmation is sent to the customer. The order is placed in the queue, updated in the database, and returned in real-time. This system assists the staff to go through the orders in real-time and process them efficiently with minimal errors.

**1.** **Introduction**

The labour rates are increasing steadily year on the year thus making it difficult to find employees. The food industry is highly labour intensive and the biggest expense in the food industry is the cost of employing the right kind of people to do the work. One of the ways to reduce this expense is to use modern technology to replace some of the jobs done by human beings and make machines do the work. Here we propose an “Online Food Ordering System” that has been designed for Fast Food restaurants, Take-Out or College Cafeterias. The system can also be used in any food delivery industry. This simplifies the process of food ordering for both the customer and the restaurant, as the entire process of taking orders is automated.

**Overview**

Online food delivery is a service that allows the user to order food from the desired food outlet via the internet. This can be done either by going directly to the website and placing an order or by using a mobile phone application. The introduction of an online food delivery system has been a convenient addition, which has not only reduced long queues but has also decreased the waiting time for ordered food delivery. The online food delivery system has already been adopted throughout the globe and its performance has been relatively good. The key players in the industry have been relying on partnerships and acquisition as prominent strategies to help boost their growth in the market.

**Objective**

The objective of Project on Online Food Ordering System: The main objective of the Project on Online Food Ordering System is to manage the details of Food items, Categories, Customers, Orders, Confirm orders. It manages all the information about Food items, Payments, Confirms orders, Food items. The project is built at the administrative end and thus only the administrator is guaranteed the access. The project aims to build an application program to reduce the manual work for managing the Food Item.

The main objective of the Project on Online Food Ordering System is to manage

the details of Food items, Categories, Customers, Orders, Confirm orders. It manages all the

information about Food items, Payments, Confirms orders, Food items. The project is totally

built at the administrative end and thus only the administrator is guaranteed the access. The

project aims to build an application program to reduce the manual work for

managing the Food Item, Category, Payment, Customer. It tracks all the details about

the Customer, Order, Confirms Order

The main objective of the Project on Online Food Ordering System is to manage

the details of Food items, Categories, Customers, Orders, Confirm orders. It manages all the

information about Food items, Payments, Confirms orders, Food items. The project is totally

built at the administrative end and thus only the administrator is guaranteed the access. The

project aims to build an application program to reduce the manual work for

managing the Food Item, Category, Payment, Customer. It tracks all the details about

the Customer, Order, Confirms Order

The main objective of the Project on Online Food Ordering System is to manage

the details of Food items, Categories, Customers, Orders, Confirm orders. It manages all the

information about Food items, Payments, Confirms orders, Food items. The project is totally

built at the administrative end and thus only the administrator is guaranteed the access. The

project aims to build an application program to reduce the manual work for

managing the Food Item, Category, Payment, Customer. It tracks all the details about

the Customer, Order, Confirms Order

The main objective of the Project on Online Food Ordering System is to manage

the details of Food items, Categories, Customers, Orders, Confirm orders. It manages all the

information about Food items, Payments, Confirms Orders for Food items. The project is totally

built at the administrative end and thus only the administrator is guaranteed the access. The

project aims to build an application program to reduce the manual work for

managing the Food Item, Category, Payment, Customer. It tracks all the details about

the Customer, Order, Confirm Order

**Scope of the Project:-**

Restaurants can offer electronic ordering both through their online web or mobile site

and through sites that serve various restaurants, and all restaurants also accept orders via text

message moreover the credit point and discount coupon out that many restaurants increased

sales level as a result of accepting electronic orders. The restaurant now the day has an interactive and

up-to-date menu with all available options in an easy-to-use manner. Younger

consumers were more likely to have used online food ordering is essentially adoption on self-

service approaches. Well-designed self-service ordering systems give customers actual

control over the pace of their transaction and allow them to limit the amount of personal

the interaction of restaurant. In most cases, an increased level of control has been shown to lead

to a higher level of customer satisfaction and greater intent to use or recommend suggested the

service. The perceived convenience of a self-service system also leads to an increase in adoption

and satisfaction. In this instance, the definition of convenience is related primarily to access

convenience and transaction convenience. A customer will search for a favourite restaurant

base on customer location, choose from available items. Payment can be amongst others

either by credit card or cash.

TECHNOLOGY USED

* HTML: Page layout has been designed in HTML.
* CSS CSS has been used for all the designing parts.
* Angular CLI: Command-line interface tool that we use to initialize.
* Bootstrap: Used for designing.
* JSON: Pseudo database.

**2**. **System Overview**

The “Online Food Delivery System” should support basic functionalities for all below-listed users.

* Admin Module: Used for managing all the food items.
* Users Module: Used for ordering food.
* Login Module: Used for managing the login details.
* Contact us: Used if the customer is having complaints.
* Cart Module: Used for managing the details of Cart.
* Payment Module: It is used for managing the payment details.
* Feedback: Used for feedback details of the customers.

**2.1 Authentication & Authorization**

**2.1.1** **Authentication**:

Any end-user should be authenticated using a unique user Id and password.

**2.1.2** **Authorization**

The operations supported and allowed would be based on the user type. For example, the Administrator has the right to add product information and view customer details. He can also view order details and purchase details for food items.

Whereas the User/Buyer has a right to Add, Remove and Clear all the products from the cart.

|  |  |  |
| --- | --- | --- |
| **2.2.Environment:** |  |  |

#### SOFTWARE REQUIREMENTS:

Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

* JSON Server

* Node Version
* Angular CLI
* Postman

**3.** **Sub-system Details**

The E-Medicare System is defined, where all users need to log in successfully before performing any of their respective operations.

Find below tables that provide functionality descriptions for each type of user/sub-system. Against each requirement, indicative data is listed in column ‘Data to include’.

**3.1 Admin**

The administrator as a user is defined to perform the below-listed operations after successful login.

|  |  |  |
| --- | --- | --- |
| Objects | Operations | Data to include |
| Food items | Add  View  Delete  Search | Food, Price, Offer, Description |
| Users | View | Name, Mobile No, Email Id, Password |

**3.2 Customer**

The customer as a user is defined to perform the below-listed operations after successful login

|  |  |  |
| --- | --- | --- |
| Objects | Operations | Data to include |
| User | Register  Login | Name, Mobile No, Email Id, Password |
| Food items | Add to Cart.  Delete from Cart.  Delete all products from the cart. | Food Names, Price |

**3.3 Login | Logout**

* Go to the Registration screen when you click on the Register link.
* Go to Success screen when you login successfully after entering valid username & password fetched from the database.
* Redirect back to the same login screen if username & password are not matching.
* Implement Session tracking for all logged-in users before allowing access to application features. Anonymous users should be checked unless explicitly mentioned.

**3.4 Password Rules:**

When it comes to password safety, the longer and more complex the password is, the better. We think it’s good practice to enforce certain minimum requirements when asking users to create a new password. Of course, you have to find a balance between these requirements and user experience. I you make the sign-up process too tedious; you could be driving users away. To enforce password strength, you should define a set of rules that a password must satisfy and then enforce these with form validation.

Example password strength rules:

* + - Password strength meter
    - At least one upper case
    - At least one special character

**4. Technology stack:**

* JSON

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | **METHODS** | **Description** | **Format** |
| http://localhost:3000/admin | GET | Get all the food items | JSON |
| http://localhost:3000/admin | GET | Add a single food item | JSON |
| http://localhost:3000/admin | POST | Update food items | JSON |
| http://localhost:3000/admin | DELETE | Delete food items | JOIN |

**5. Assumptions:**

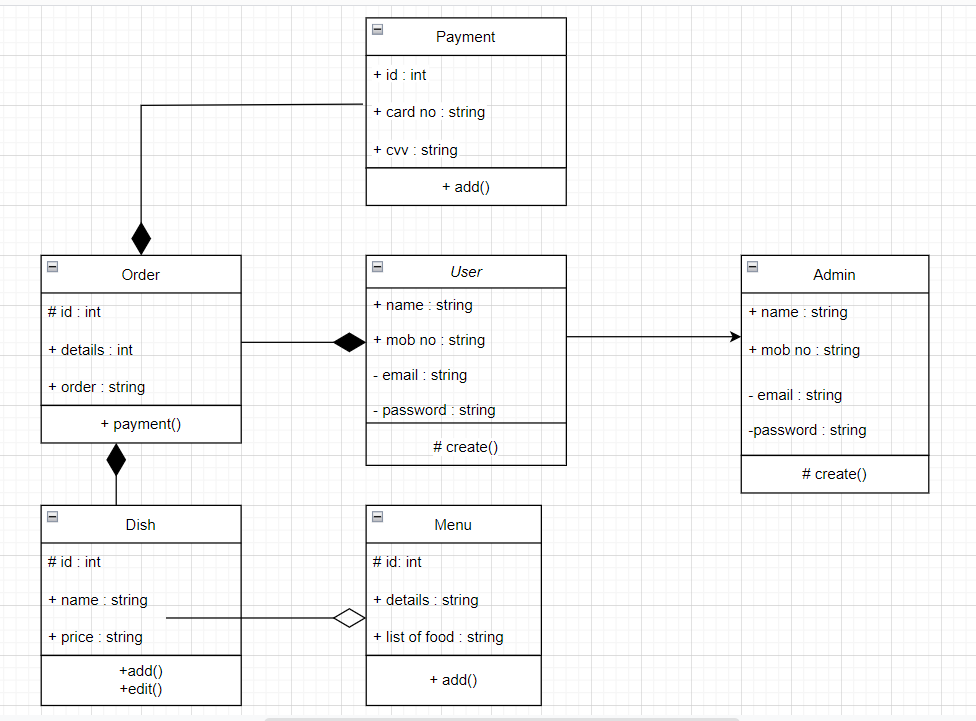
* User Interface: The type of client interface (front-end) to be supported - Angular based

* The administrator can add and remove food items into the database.
* User doesn’t have access to add food items page.

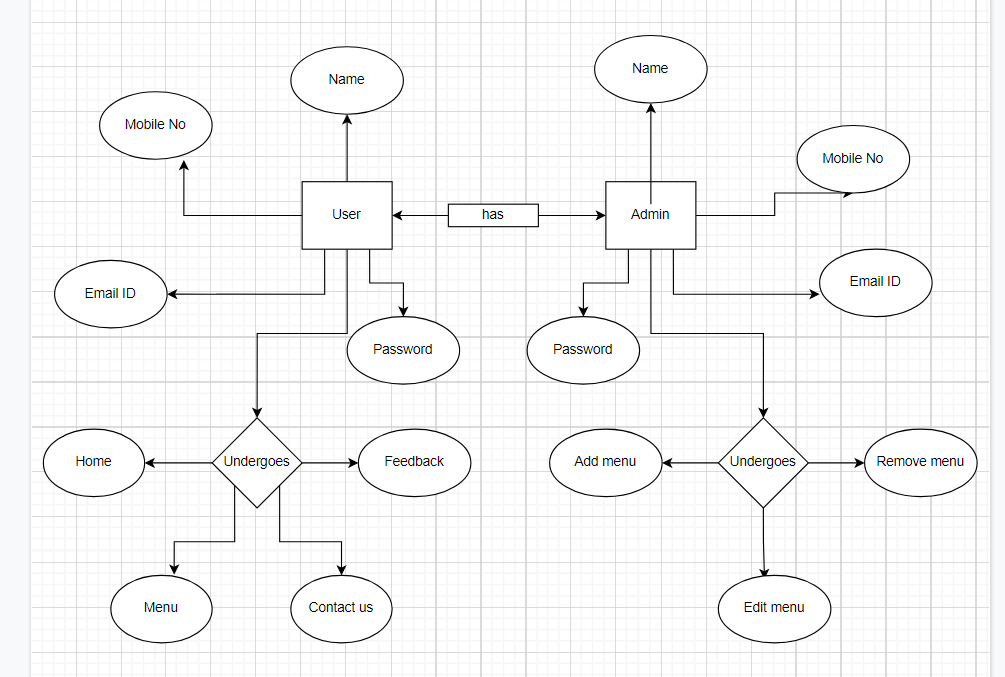
* When you add food items into the cart the No. of food items selected will be incremented.
* If you remove the food items from the cart, the counter will be decremented.
* If we remove all the food items the card will display empty.
* The total amount will be calculated based on the food items, accordingly, change the food items counter & total amount.

**6. Diagrams:**

6.0.1 Class Diagram:



**6.0.2 ER-Diagram:**

****

**6.1** **Unit testing:**

JUnit is a unit testing framework for the Java programming language. It plays a crucial role in test-driven development and is a family of unit testing frameworks collectively known as xUnit.

JUnit promotes the idea of "first testing than coding", which emphasizes setting up the test data for a piece of code that can be tested first and then implemented. This approach is like "test a little, code a little, test a little, code a little." It increases the productivity of the programmer and the stability of program code, which in turn reduces the stress on the programmer and the time spent on debugging.

**Features of JUnit:**

* JUnit is an open-source framework, which is used for writing and running tests.
* Provides annotations to identify test methods.
* Provides assertions for testing expected results.
* Provides test runners for running tests.
* JUnit tests allow you to write codes faster, which increases quality.
* JUnit is elegantly simple. It is less complex and takes less time.
* JUnit tests can be run automatically and they check their results and provide immediate feedback. There's no need to manually comb through a report of test results.
* JUnit tests can be organized into test suites containing test cases and even other test suites.
* JUnit shows test progress in a bar that is green if the test is running smoothly, and it turns red when a test fails.

**Unit Test Case**

A Unit Test Case is a part of code, which ensures that another part of code (method) works as expected. To achieve the desired results quickly, a test framework is required. JUnit is a perfect unit test framework for the Java programming language.

A formal written unit test case is characterized by a known input and an expected output, which is worked out before the test is executed. The known input should test a precondition and the expected output should test a post-condition.

There must be at least two unit test cases for each requirement − one positive test and one negative test. If a requirement has sub-requirements, each sub-requirement must have at least two test cases positive and negative.

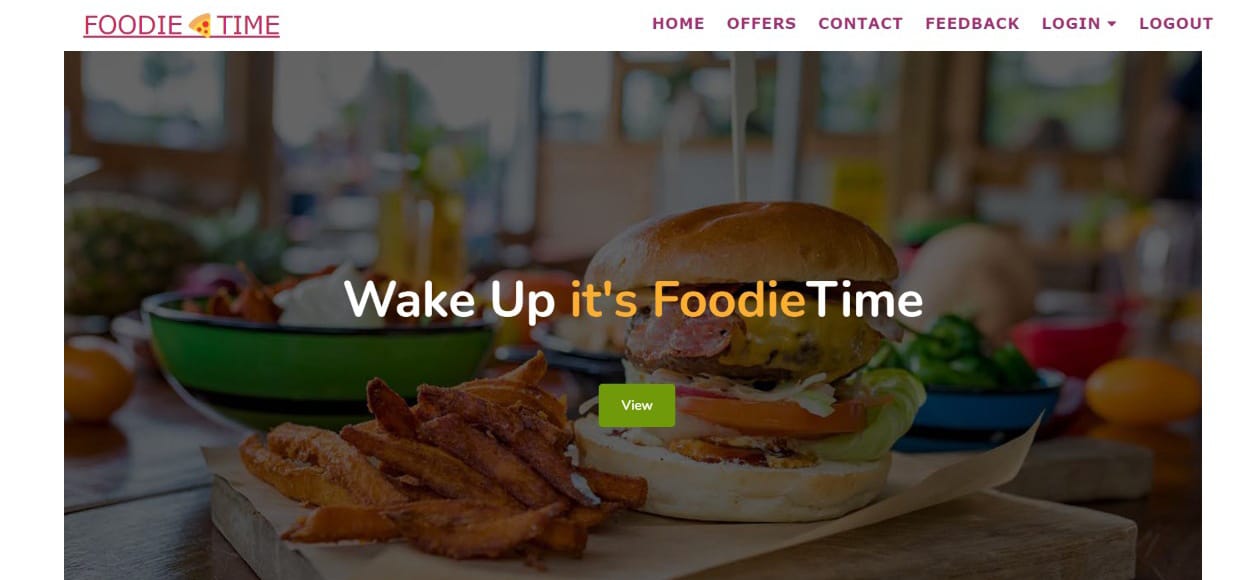
**6.2 Functional Testing using POSTMAN tool:**

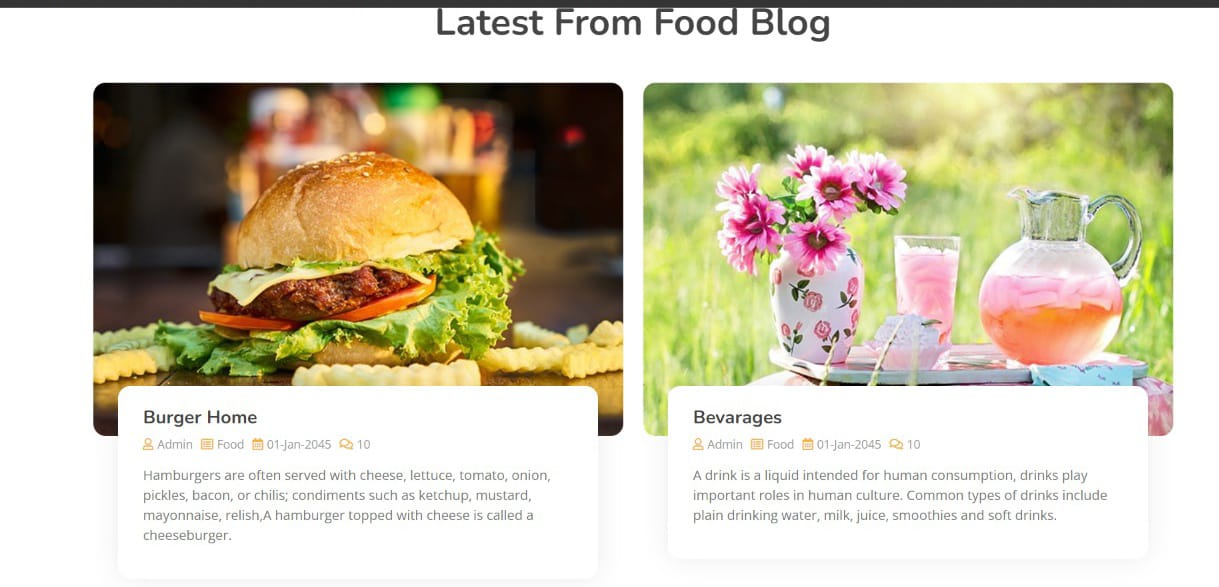
Tests are automated by creating test suites that can run again and again. Postman can be used to automate many types of tests including unit tests, functional tests, integration tests, end-to-end tests, regression tests, mock tests, etc. Automated testing prevents human error and streamlines testing.

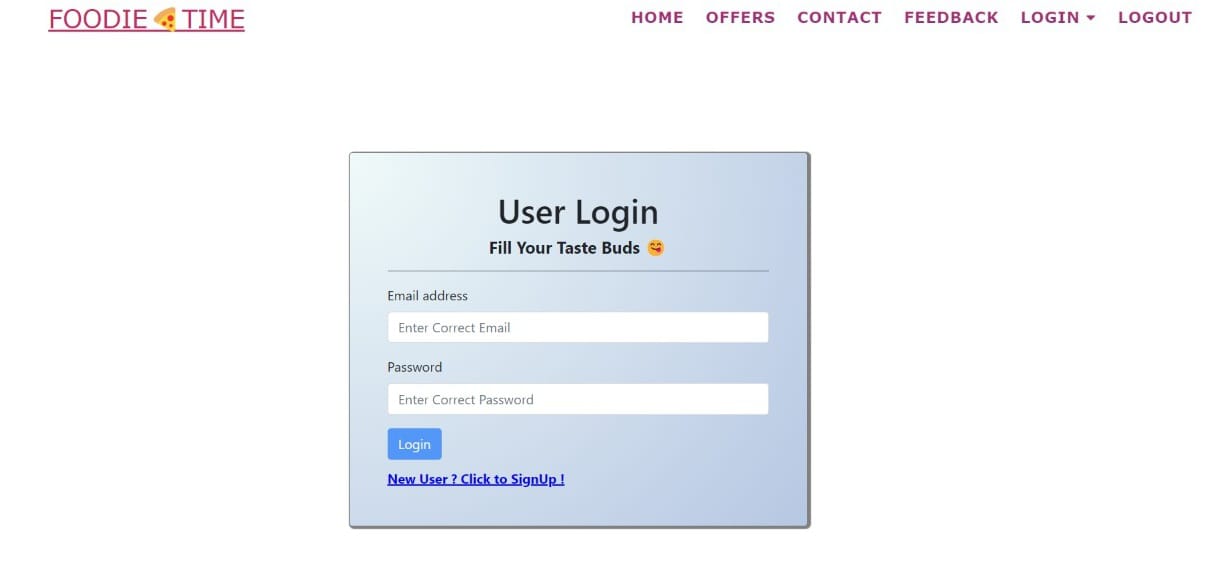
**7. Advantages**

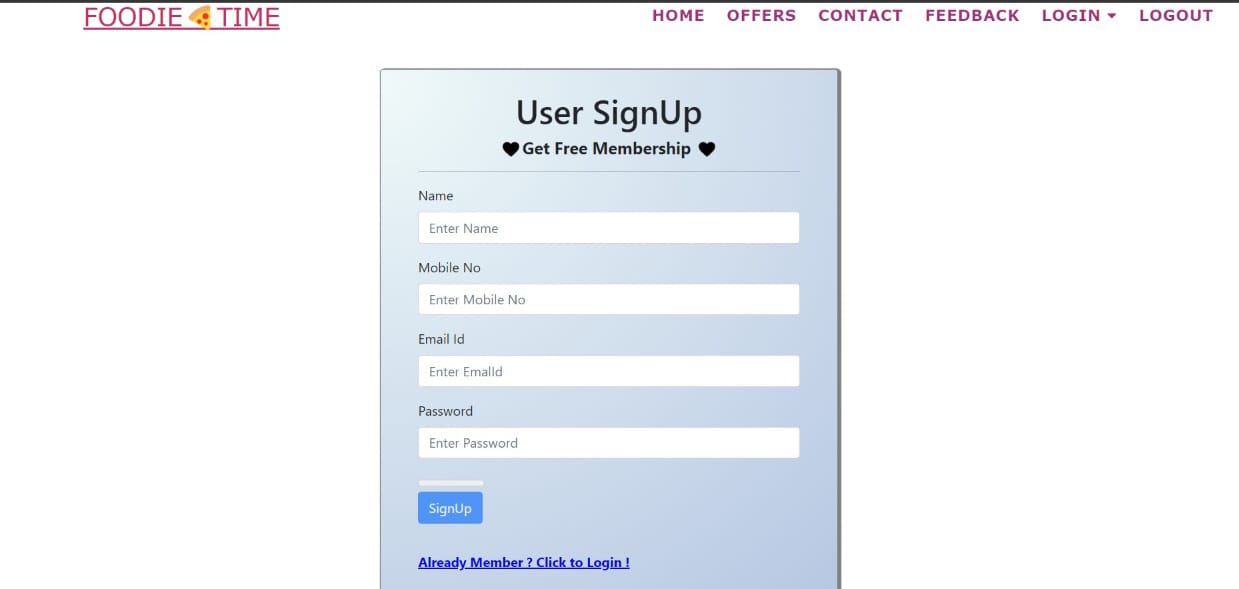
* Easy communicative.
* Time-saving.
* Always open the restaurant.
* Payment.

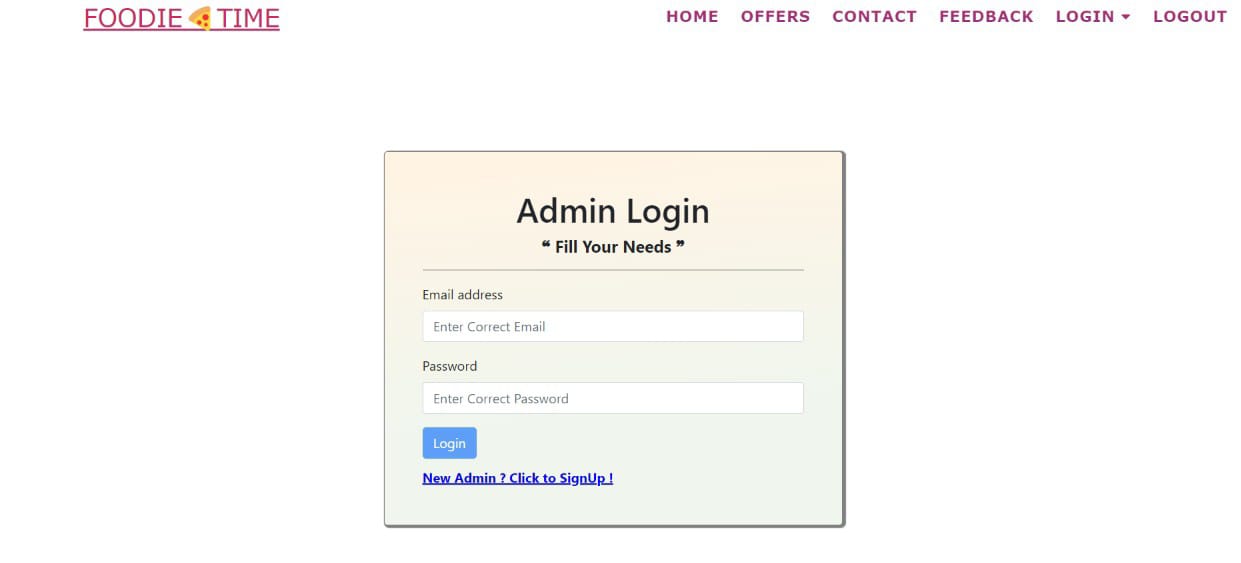
**8. Output Screenshots:**

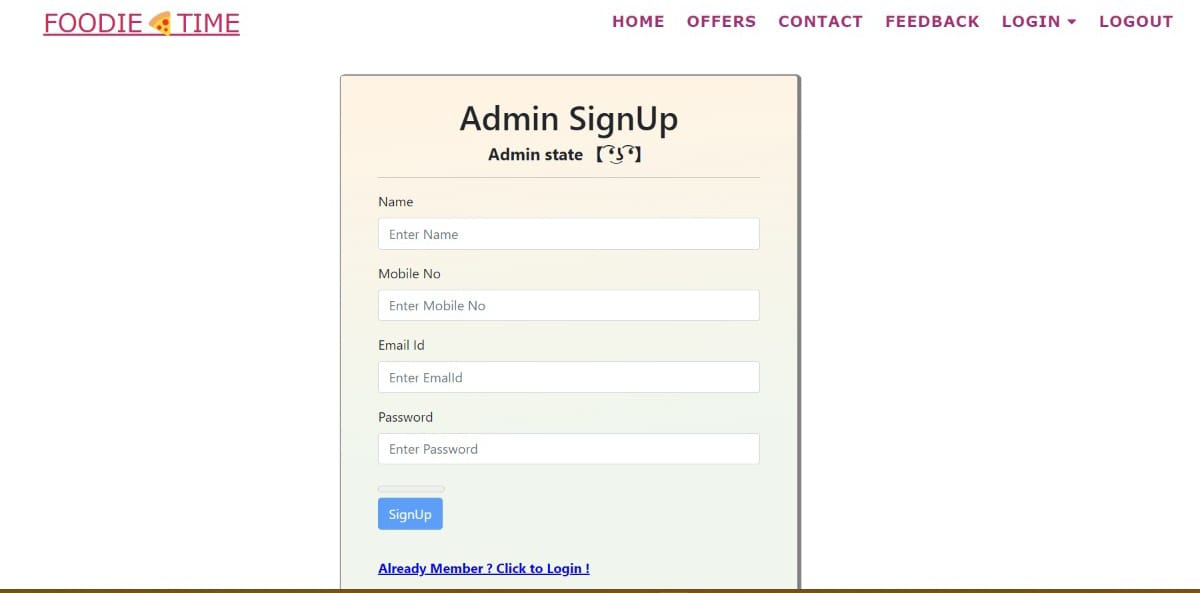


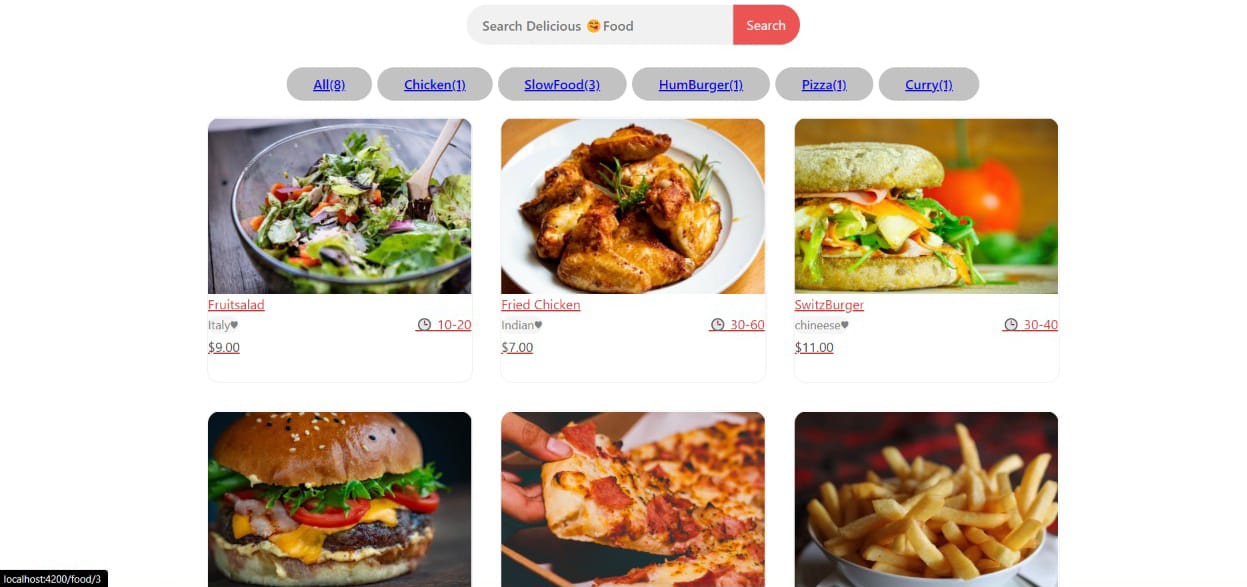


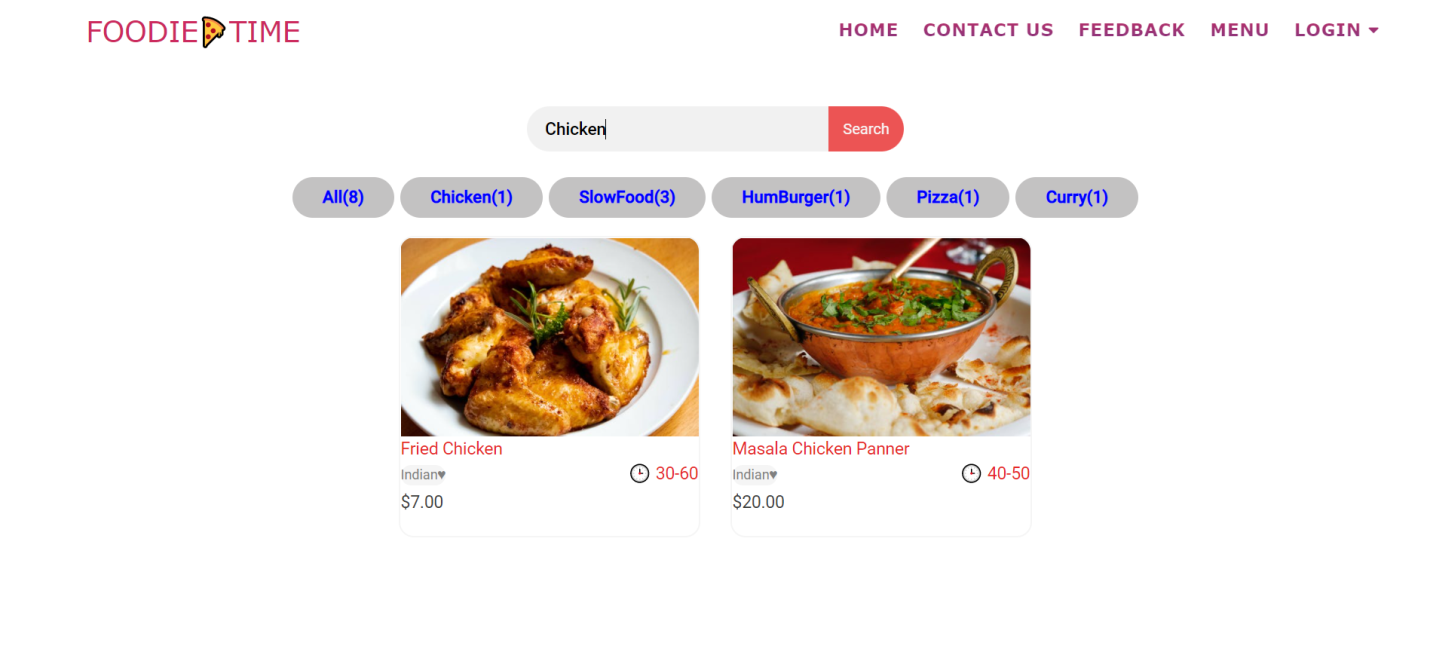


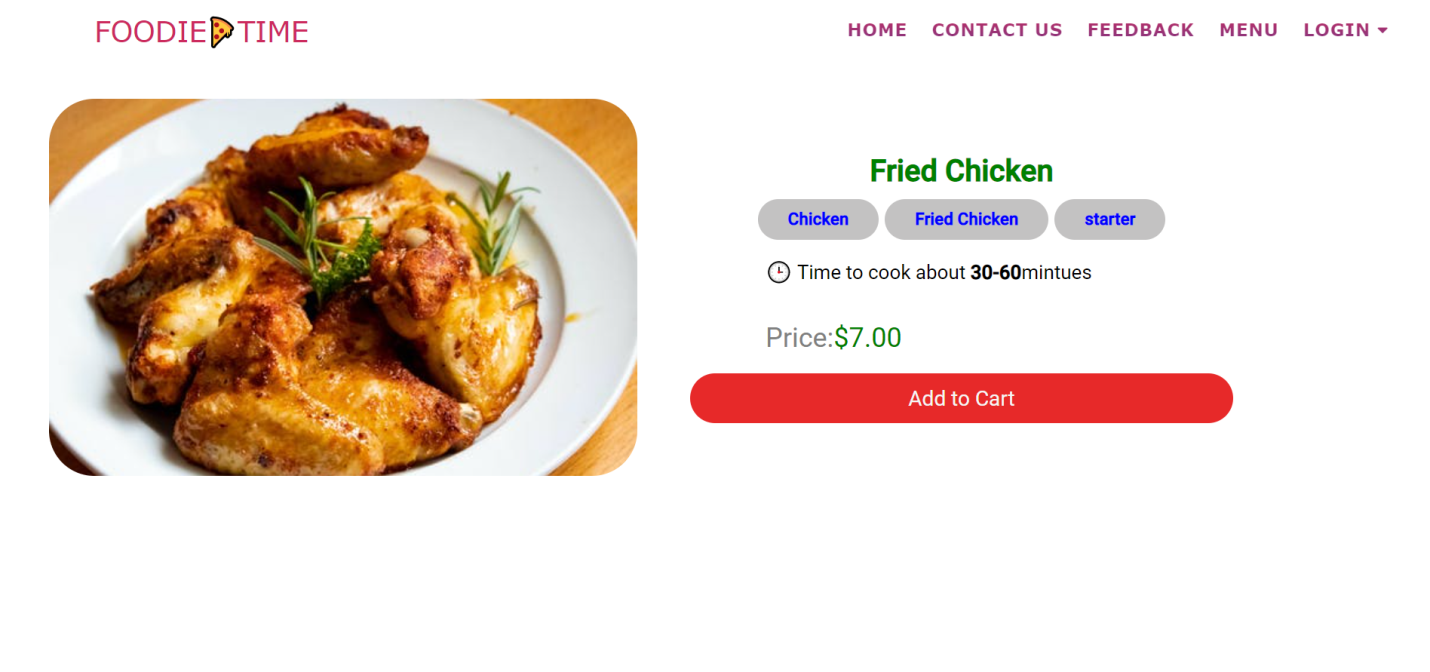


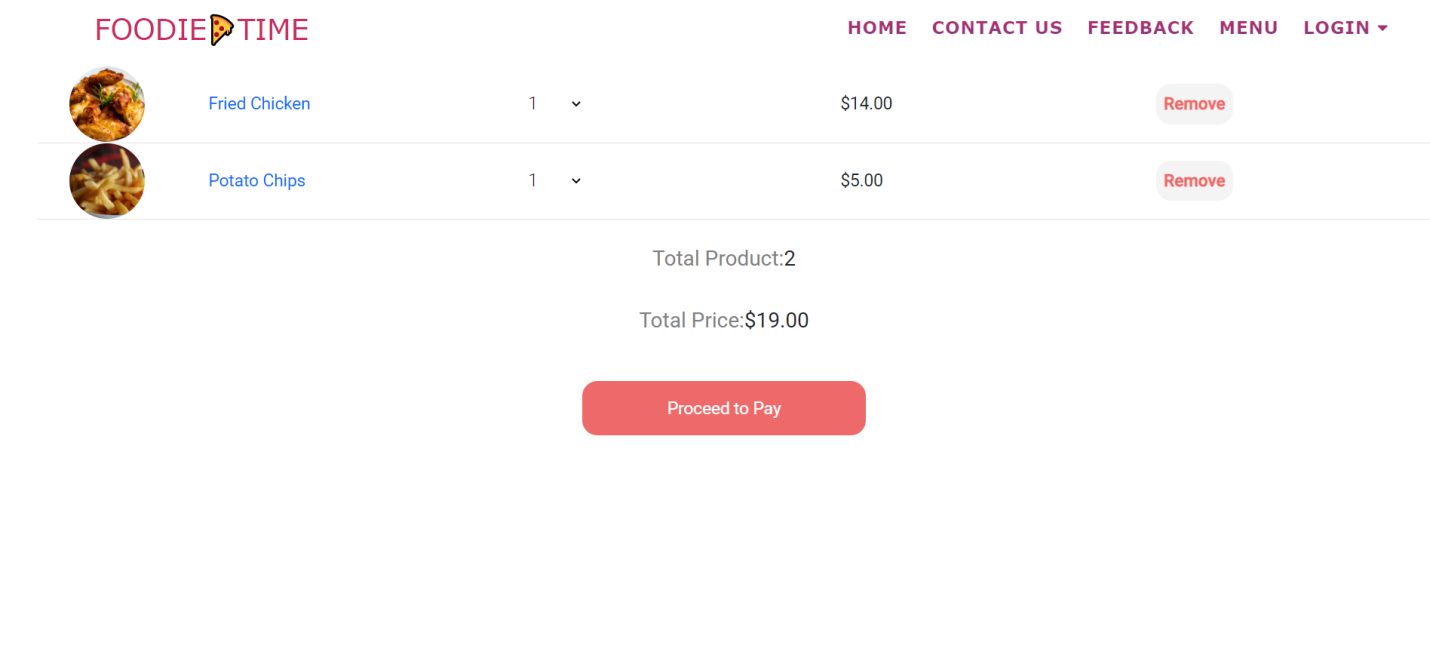


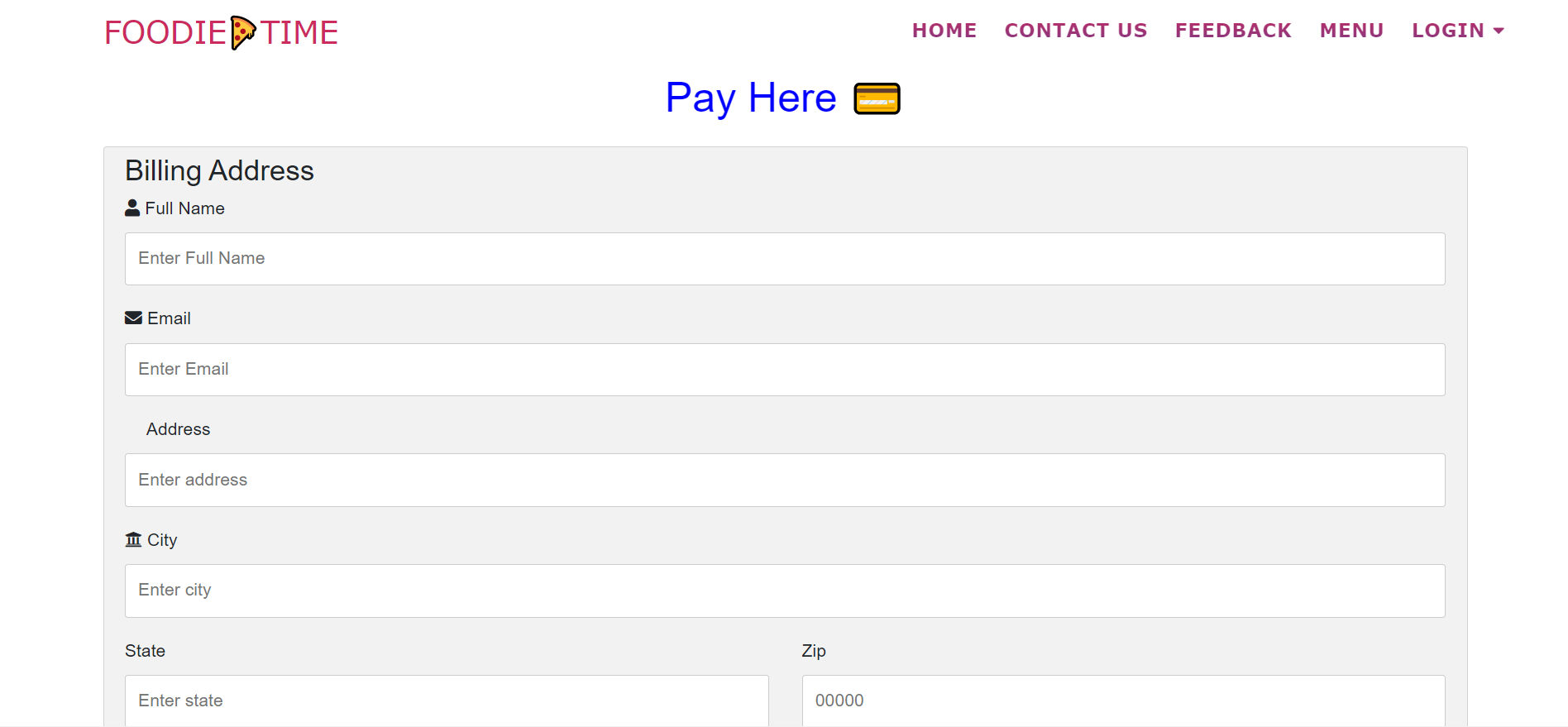


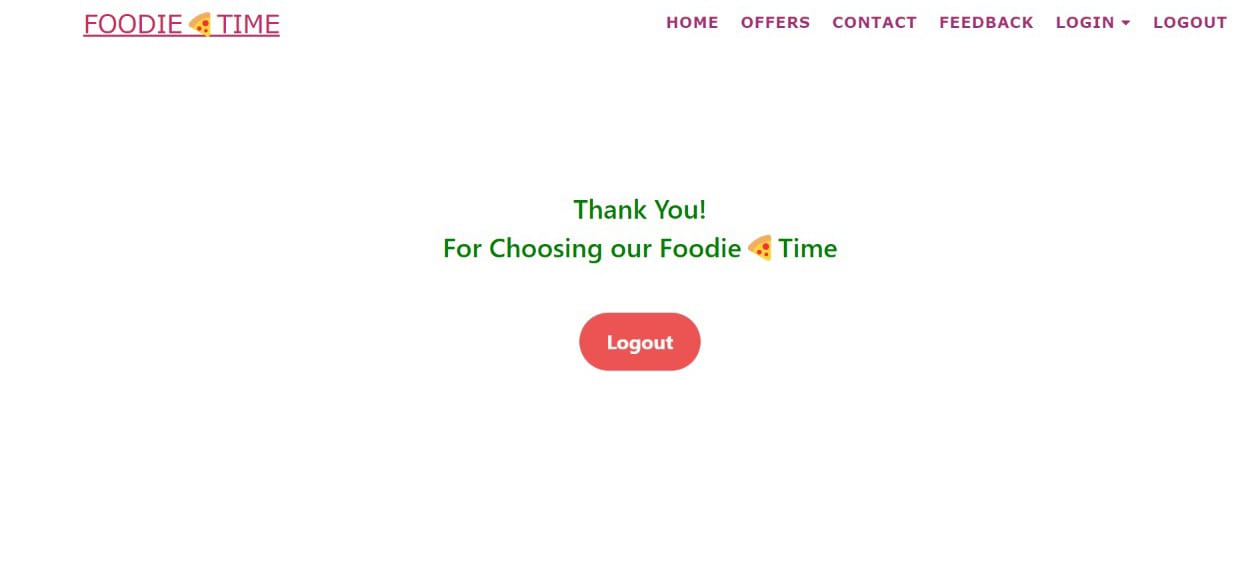


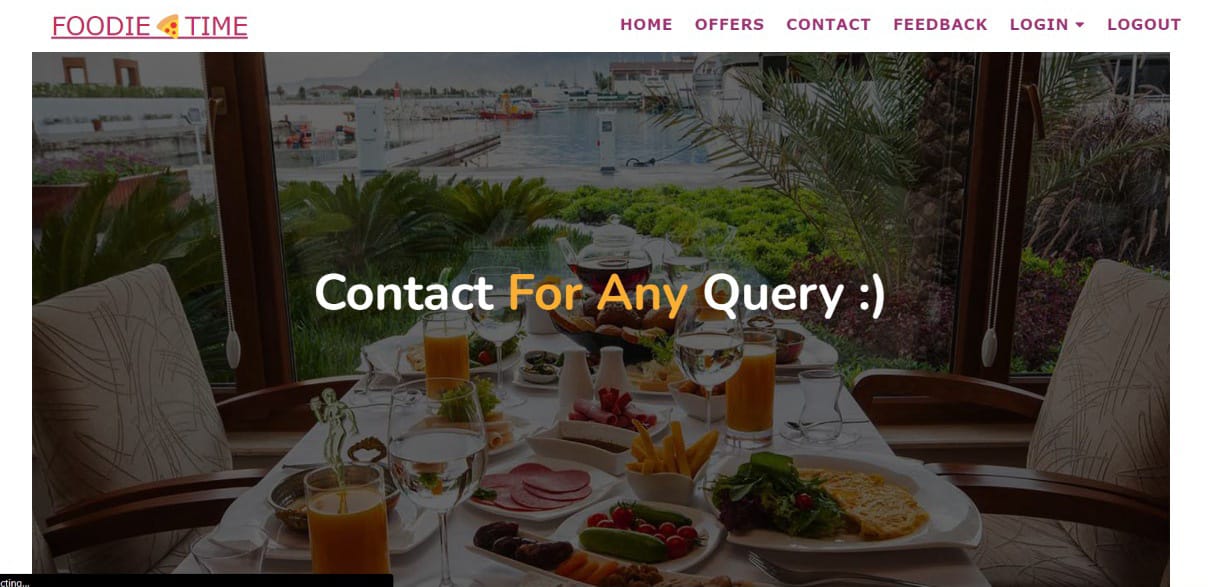


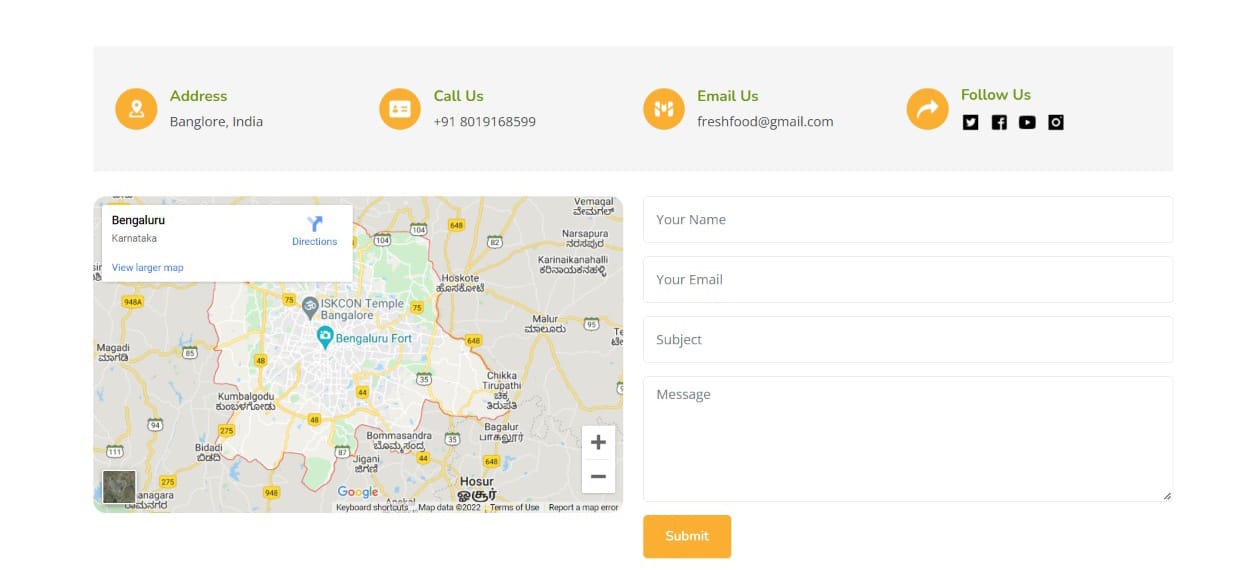


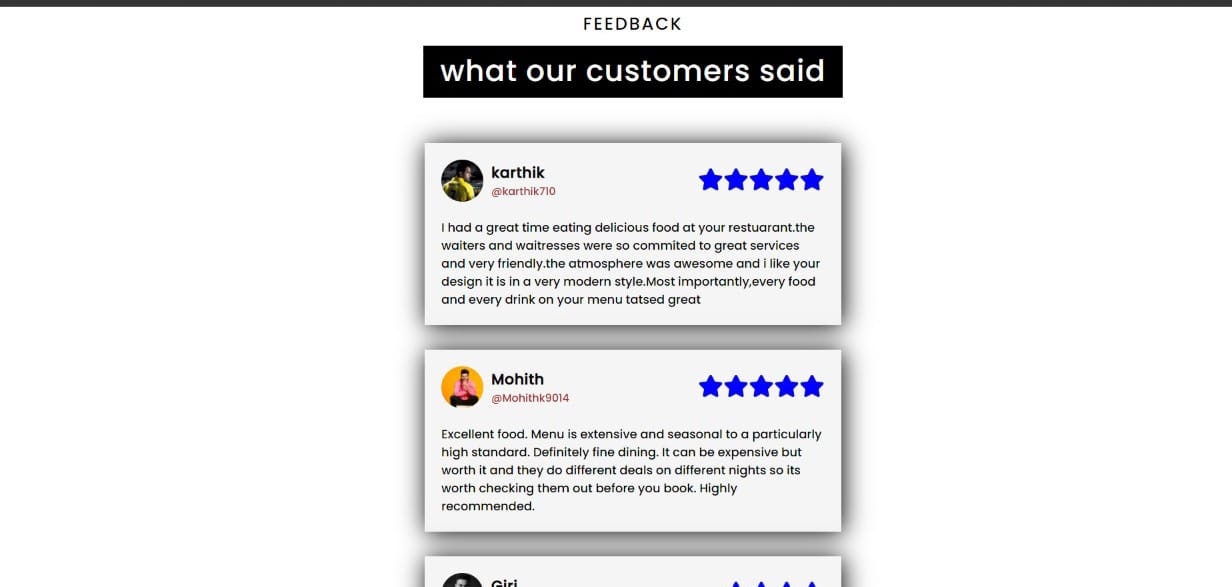












**9. CONCLUSION**

An online food ordering system is developed where the customers can make an order for the food and avoid the hassles of waiting for the order to be taken by the waiter. Using the application, the end-users register online, read the E-menu card and select the food from the e-menu card to order food online. Once the customer selects the required food item the chef will be able to see the results on the screen and start processing the food. This application nullifies the need for a waiter or reduces the workload of the waiter. The advantage is that in a crowded restaurant there will be chances that the waiters are overloaded with orders and they are unable to meet the requirements of the customer satisfactorily. Therefore by using this application, the users can directly place the order for food to the chef online. In conclusion, an online food ordering system is proposed which is useful in small family-run restaurants as well as in places like college cafeterias, etc. This project can later be expanded on a larger scale. It is developed for restaurants to simplify their routine managerial and operational task and to improve the dining experience of the clients. This also helps the restaurant owners develop healthy customer relationships by providing reasonably good services. The system also enables the restaurant to know the items available in real-time and make changes to their food and beverage inventory based on the orders placed and the orders completed.